Sinclair

ZX SPECTRUM

Userclub

Clubinfo Nr. 3 (108) vom 01. März 1993

Rolf Knorre - Postfach 20 01 02 - 5600 Wuppertal 2
Telefon 0202 / 55 66 89 - Telefax 0202 / 59 47 21
Stadtsparkasse Wuppertal (BLZ 33050000) Kto. 853382

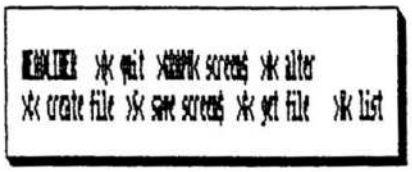
Hallo Freunde,

da auch diesen Monat nicht genug Beiträge eingegangen sind, habe ich mal wieder in alten Ausgaben geblättert und daraus etwas wiederveröffentlicht. Da wir viele neue Mitglieder haben, die zu dieser Zeit noch nicht im Club waren, dürfte daran auch Interesse bestehen. Trotzdem hoffe ich noch immer auf eine stärkere Mitarbeit. Viel Spaß mit dem neuen Info.

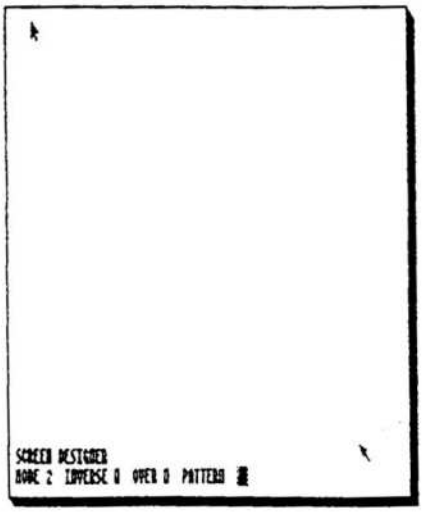
Rolf Knorre

P.S. Bitte teilt mir nach und nach die neue Postleitzahl mit, damit ich meine Datei umstellen kann. Danke.

COL! - TEIL 2: Wie versprochen geht es heute um die Merge - Funktion von Col!. **MERGE:** Als erstes wird zusätzlich der Headliner! in den Speicher von WM geladen. Um mit Col! -



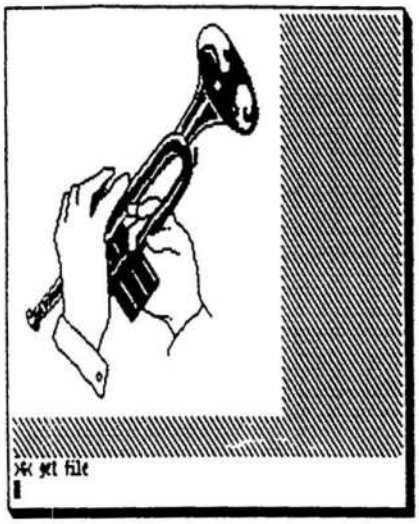
Merge arbeiten zu können müssen wir uns ein >**pattern**<-File erstellen. Mit



G(et) "Name" und Enter wird der Headliner aktiviert, es erscheint ein neues (Bild 1) Menue. Nun Taste "A" (Alter) drücken, wir konzentrieren uns auf die Pattern-Funktion. Der Headliner! wird noch in unserer DTP -

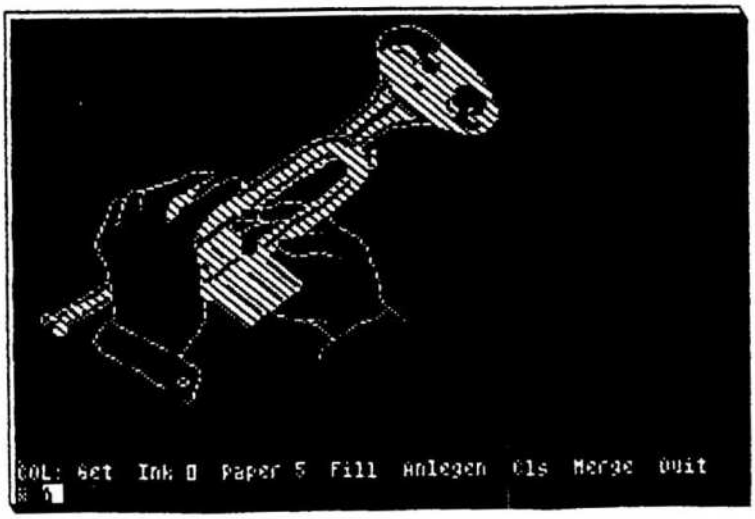
Serie ein Thema sein. Als nächstes gehen wir in den "Grafik-Modus" und wählen (Taste A > U) unser bevorzugtes Muster (Bild 3). Nun Taste "F" (Fill) drücken, der Bildschirm füllt sich mit dem zuvor ausgesuchten Pattern - Muster. INV.VIDEO drücken und den Screen anlegen (Taste "C") bzw. saven. Mit der Funktion "Quit" könnt Ihr nun aus dem Headliner aussteigen. Das Programm Headliner! wird nun vorläufig nicht mehr gebraucht. Col! in den Speicher von WM laden und mit G(et) und Enter aktivieren. Mit der G(et)-Funktion wird nun das erstellte Headliner - Muster aufgerufen. Jetzt mit Taste "M" die Merge-Funktion aktivieren und den Namen des Grafikfiles (Bild 4) eingeben. Gleich im Anschluß daran wird der Name des Headliner-Files eingegeben und Enter. Col! übernimmt nun die Arbeit, Bild 5 (unten) zeigt das Ergebnis dieser Aktion. Ihr könnt nun den "neuen" Screen anlegen bzw. absaven. Mit der Ink- bzw. Paper-Funktion kann nun noch hin und her getestet werden. ***** ENDE *****

Fazit: Das Programm Col!



ist für Wordmaster und seine Fans eine echte Bereicherung. Im nächsten Teil geht es um die Doppel- und Mehrfachfonts "EXOUT 2" sowie "CIRCUS 3", aber bis dahin wünsche ich Euch noch viel Spaß mit DTP LG Teil 16. ■

Oldenburg, den 04. 02. 93



Ludwig Halbritter
Augsburger Str. 4
8935 Fischach
Tel.:08236/495

Fischach, 21.02.93

Hallo Spectrumfreunde,

heute möchte ich mit "BENZIN", lfd.-Nr. 300, ein weiteres Programm aus der PD-Bibliothek vorstellen. Es wurde von H. Surrer geschrieben und auf der Clubkassette Nr. 7 des SUCW weitergegeben.
Das Programm "BENZIN" gibt jedem die Möglichkeit, den Verbrauch und die Kosten des Kraftstoffes für sein Auto jederzeit zu kontrollieren.

Nach dem Laden des Programmes wird zuerst das Berichtsjahr und der Kilometerstand des Autos eingegeben. Nach dem nächsten Tanken wird dann der Monat, der aktuelle Kilometerstand, die getankte Menge und die Kosten eingetragen. Bei jedem weiteren Tankstop werden die Daten ebenfalls eingegeben.

Die tabellarische Ausgabe der Daten liefert folgende Informationen.

* BENZINKOSTEN FUER A-ZE 73

Mon	STAND	KM	LITER	DM	DM/L
Jan	119134	864	69.5	92.80	1.335
Feb	119797	663	54.3	73.40	1.352
Mae					
Apr					
Mai					
Jun					
Jul					
Aug					
Sep					
Okt					
Nov					
Dez					
'93	GESAMT	1527	123.8	166.20	1.342

(W) = Weiter / (M) = Menue
(C) = Hardcopy gesamt

Es wird für jeden Monat der letzte Kilometersatnd, die gefahrenen Kilometer, der Gesamtverbrauch, die Gesamtkosten und der durchschnittliche Preis je Liter angezeigt.

Weiterhin können die Kosten je gefahrenen Kilometer, die Differenz zum Durchschnittswert, der Verbrauch auf 100 Kilometer und die Differenz zum Durchschnittsverbrauch abgerufen werden.

IM JAHRE 1993 *

Mon	PF/KM	DIFF	VERBR	DIFF
Jan	10.7	-0.1	8.04	-0.06
Feb	11.1	+0.2	8.19	+0.08
Mae				
Apr				
Mai				
Jun				
Jul				
Aug				
Sep				
Okt				
Nov				
Dez				
'93	10.9	>>>>>	8.11	<<<<<

(W) = Weiter / (M) = Menue

Außerdem kann graphisch der Durchschnittsverbrauch je Monat in einem Balkendiagramm dargestellt werden.
Zum Speichern der Daten gibt es eine SAVE-Option für Kassette und Microdrive.

Manfred Messerschmidt, Wildtaubenweg 64, 7050 Waiblingen, ☎ 07151/22387

AUSGETRICKST

Nun, wer wurde ausgetrickst? Fangen wir von Anfang an. Vermutlich kennt jeder Masterfile. Ein sehr gutes Datenverarbeitungprogramm. Wer schon mit ihm arbeitete, der kennt seine Vorzüge. Der Druckertreiber war nur für den Sinclair Drucker ausgelegt. Als die Möglichkeit bestand am Spectrum einen richtigen Drucker anzuschließen, kam zusätzlich ein neuer Druckertreiber hinzu. Äußerst praktisch wurden die Adressen für den Druckertreiber (59770-61512) gewählt. Sobald er geladen ist überblendet er vom Masterfile den Editor. Das Editormenue wird nur für den Maskenlayout benötigt. Separat zum Druckertreiber braucht man noch ein Druckerlayoutgenerator. Mit dem Programm wird extern das Druckbild gestaltet. Jedes einzelne Datenfeld läßt sich anwählen und die Druckposition exakt bestimmen. Neben der Textpositionierung kann pro Datenfeld noch zusätzlich 10 Druckerkontrollcodes mit übermittelt werden.

In regelmäßigen Abständen drucke ich für einen Verein Adressaufkleber. Bis auf eine kleine Unregelmäßigkeit war ich mit dem Druckbild zufrieden. Sobald zwei unterschiedliche Datenfelder mit Druckerkontrollcodes in einer Linie stehen werden sie ignoriert. Hier ein Beispiel:

Herr
Dagobert Maus
Dschungelweg 77
3333 Entenhausen

Konzentrieren wir uns auf die letzte Zeile. Die Postleitzahl sowie die Ortsangabe kommen aus unterschiedlichen Datenfeldern. Soll die Postleitzahl in Breitschrift und die Ortsangabe in Normalschrift gedruckt werden, dann setzt man normalerweise im Drucklayoutgenerator jeweils die Druckerkontrollcodes. In diesem Fall heben sich die beiden Codes gegenseitig auf. Der Ausdruck erscheint in Normalschrift. Sobald nur ein Datenfeld in Breitschrift angewählt ist, druckt er die ganze Zeile breit. Zuerst war ich der Meinung, es liegt an der Druckart. Denn mein Druckkopf vom Drucker arbeitet in beiden Richtungen. Über Druckerkontrollcodes generierte ich den UNIDIREKTIONALER AUSDRUCK. Das heißt nur von links nach rechts ist der Druckkopf aktiv. Auch das brachte keine Besserung. Dann ging mir ein Licht auf. Die Ortsangabe im Drucklayoutgenerator kommt in die fünfte anstatt vierte Zeile. Durch Druckerkontrollcodes den Papierrücktransport aktivieren. Jetzt zieht der Drucker das Papier in die vierte Zeile, bevor der Druckkopf zu arbeiten beginnt. Die Breit- bzw. Normalschrift steht nun in einer Zeile.

Zwei Dinge sollte man beachten:
- Sobald unterschiedliche Schriftbreiten in einer Zeile sind, muß auf die Überlagerung bzw. auf zu große Abstände geachtet werden. Bei Masterfile kann die Datenfeldlänge variiern. Recht gut eignet sich die Postleitzahl, denn sie hat im Moment eine fixe Länge von vier Zeichen.
- Der Rücktransport funktioniert nur bei Endlospapier.

[illegible] ********** Harald Lack
Kirchdorfer Str. 5 * DOUBLE 'H' SOFTWARE * Heidenauer Str. 5
8201 Großholzhausen ********** 8201 Raubling

Hallo Freunde!!!

Nachdem im Club bereits des Öfteren der Ruf nach Programmbeschreibungen bzw. Plänen laut wurde, haben wir uns entschlossen, wieder einmal ein Game zu "verplanen" und mit einer Beschreibung zu versehen. Wie vielleicht der eine oder andere schon bemerkt hat, sind wir Fans von Oldies. Einer dieser Oldies ist u. a. Freddy Hardest Teil II mit dem wir uns beschäftigt haben. Für die Erstellung des Planes haben wir mit folgenden Grafikprogrammen gearbeitet: DRAW I, ARTSTUDIO und THE ARTIST in Verbindung mit WORDMASTER.

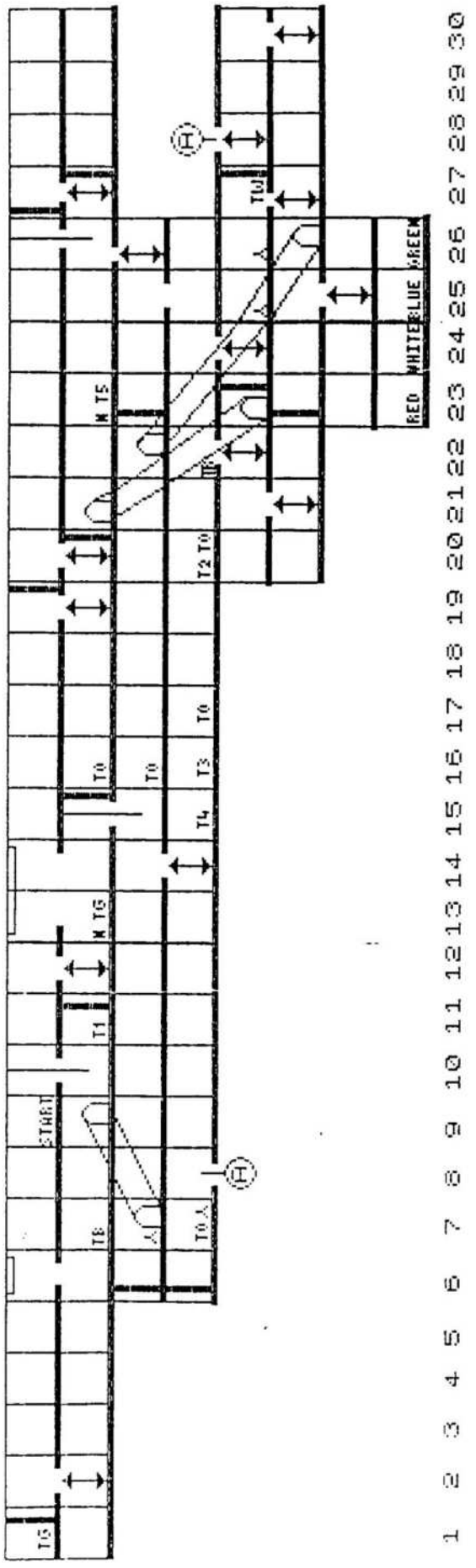
Nachdem Freddy Hardest II allgemein bekannt sein dürfte, geben wir nur eine kurze Beschreibung der Handlung. Ziel des Spiels ist es, mit einem der vier zur Verfügung stehenden Raumschiffe vom Planeten zu fliehen. Vor der Flucht muß die Kiste natürlich aufgetankt werden (mittels einer Nuklearzelle an dem entsprechenden Terminal = T5). Diese Nuklearzellen liegen verstreut, nach jedem Programmstart an einem anderen Platz, in dem Koplex. Sie sind als blinkendes Quadrat mit dem Zeichen für Radioaktivität dargestellt und werden durch einfaches darüberlaufen eingesammelt. Abzulegen sind sie an der am Boden mit "N" markierten Stelle neben einem Nuklearterminal. Das allein hilft natürlich wenig, wenn das Raumschiff nicht auf Hyper Space gebracht wird (auch mittels Terminal). Diese Terminals sind je nach Farbe des Schiffes mit dem entsprechenden Buchstaben gekennzeichnet, z. B. TB für blaues Raumschiff. Allerdings sind einige nicht zu gebrauchen, die haben wir mit TO = Terminal out of order gekennzeichnet. Außerdem benötigt jedes Raumschiff einen bestimmten spezifischen Code den man wiederum an verschiedenen Terminals erhält.

T1 = für rotes Raumschiff = Borax
T2 = für grünes Raumschiff = Fenix
T3 = für blaues Raumschiff = Audax
T4 = für weißes Raumschiff = Artex

Die vertikalen Pfeile stellen Aufzüge dar.

Für all diejenigen, die den Teil I diese Programmes nicht haben hier der Code um den Teil II zu starten : 897653. Außerdem kann es nicht schaden, wenn man ein paar Leben mehr zur Verfügung hat, die man mit dem Poke 61607,183 erhält.

Wir wünschen euch viel Spaß beim nachspielen und verbleiben bis bald .....


1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
RED WHITE BLUE GREEN
START
T0
T1
T2
T3
T4
T5
T6
T8
T9

Maschinencode-Programme auf dem Microdrive

Viele Maschinencode-Programme sind sehr einfach auf eine Microdrive-Cassette zu überspielen, indem man mit einem Kopierprogramm Startadresse s und Länge l des Maschinencode-Blocks feststellt und dann diesen mit SAVE*"m";1;"Name"CODE s,l abspeichert. Nach LOAD wird der Maschinencode mit RANDOMIZE USR x aufgerufen. x ist dabei dem Basic-Vorspannprogramm zu entnehmen.

Dies funktioniert aber nicht bei jedem Maschinencode-Programm. Betroffen sind hier diejenigen,die auch den für das Microdrivesystem reservierten Speicherplatz einnehmen, d.h. Adressen unter ca. 25000 benutzen (z.B. Pinball). Wenn man versucht, solche Programme auf Microdrive zu SAVEn, so erscheint im besten Fall ein "Out of memory", es kann aber auch dazukommen, dass der Microdrive anläuft, aber keine Daten überträgt und nur noch durch Stromentzug zu stoppen ist (Siehe auch Leserbrief im CHIP 8/84, S.21). Dies führt dann zu Beschädigungen der auf der Cartridge gespeicherten Informationen.

Deswegen braucht man aber nicht auf die oben genannten Programme zu verzichten. Es gibt hier einen kleinen Trick, und der nennt sich Blockverschiebung. Im Prinzip funktioniert dies so: Der zu kopierende Maschinencode wird von der Cassette in eine andere, höhere Speicheradresse geladen und entsprechend mit dem Microdrive geSAVEd, was nun ohne Probleme geht. Nach erneutem Laden, diesmal vom Microdrive, befindet sich das Programm also an einer falschen Adresse. Nun wird es durch eine zusätzliche, kleine Maschinencode-Routine in praktisch null komma nichts zur richtigen Adresse verschoben, wo es dann gestartet werden kann, da jetzt ja das Microdrive-Map nicht mehr benötigt wird.

Das untenstehende Programm macht diese Prozedur relativ einfach, da es die kleine Maschinencode-Routine automatisch nach dem Hauptprogramm abSAVEt. Nach dem Eintippen kann es mit SAVE (*"m";1;) "MICROCODE" LINE 10 gespeichert werden.

Soll nun kopiert werden, so ist zuerst dieses Hilfsprogramm zu laden und die gefragten Werte sind einzugeben. Mit "neuer Startadresse" ist eine beliebige Adresse gemeint, unter der das Programm auf Microdrive gespeichert wird. Die "ursprüngliche Startadresse" und die "Länge des Blocks sind zuvor mit einem Kopierprogramm (z.B. Mr.Copy) zu ermitteln. Dann wird die Routine den eingegebenen Werten angepasst und der Speicherplatz für das nun zu ladende Maschinencode-Programm freigehalten. Das danach folgende SAVEn kann auf Microdrive("M") oder - für Testzwecke - auf Cassette ("C") vorgenommen werden.

Abschliessend erscheint die Befehlsfolge, um das Maschinenprogramm wieder zu laden und zu starten. Die Aufrufadresse ist dem Basicprogramm zu entnehmen, mit dem der Maschinencode ursprünglich von der Cassette geladen wurde. Ist dieses nicht vorhanden, d.h. starten sich die Bytes des Maschinencodes selber, so ist dieser mit dem vorliegenden Programm nicht zu kopieren.

Sollte hierfür jemand eine Lösung kennen, so bin ich für entsprechende Hinweise dankbar.

Der Aufwand lohnt sich aber auf jeden Fall, wenn man an die sekundenschnelle Ladezeit, selbst für die längsten 48K-Programme, denkt.

Albert Briner
In den Gartenhöfen 23
CH-4153 Reinach

```
 10 REM     MICROCODE
 20 REM        VON
 30 REM     A. BRINER
 40 REM
 80 PAPER 1: BORDER 1: INK 7: CLS : POKE 23609,50
100 INPUT "Neue Startadresse : ";a
110 CLEAR a-1
120 LET a=PEEK 23730-256*PEEK 23731+1
130 PRINT "Neue Startadresse : ";a
150 INPUT "Urspruengliche Startadresse :    ";b
160 PRINT '"Urspruengliche Startadr.: ";b
170 INPUT "Laenge des Blocks :";c
180 PRINT '"Laenge des Blocks : ";c
200 LET a2=INT (a/256): LET a1=a-a2*256
210 LET b2=INT (b/256): LET b1=b-b2*256
220 LET c2=INT (c/256): LET c1=c-c2*256
250 DATA 33,a1,a2,17,b1,b2,1,c1,c2,237,176,201
260 RESTORE
270 FOR i=a+c TO a+c+11
280 READ d
290 POKE i,d
300 NEXT i
400 PRINT AT 8,0;"Bitte das Band mit dem Maschinencode  einlegen, eine Tas- dr
uekken, und das Band starten!"
410 PAUSE 0: PRINT AT 8,0;"Der Maschinencode wird nun ab   Adresse ";a;" in den
 Speicher   geladen.                                    "
420 LOAD ""CODE a
430 PRINT AT 8,0;"Jetzt wird der Maschinencode zu-sammen mit der Routine fuer
dieBlockverschiebung geSAVEd."
433 PRINT TAB 30: PRINT TAB 30
435 INPUT "Name fuer SAVE? ";n$
437 PRINT "Name : ";n$
440 PRINT ' INVERSE 1;"C"; INVERSE 0;"assette oder "; INVERSE 1;"M"; INVERSE 0;
"icrodrive?"
450 PAUSE 0
460 IF INKEY$="C" OR INKEY$="c" THEN  LET q=0: LET m$=" von  Cassette": PRINT A
T 15,0;TAB 30: SAVE n$CODE a,c+12: GO TO 500
470 IF INKEY$="M" OR INKEY$="m" THEN  LET q=7: LET m$="vom Microdrive": PRINT A
T 15,0;TAB 30: SAVE *"m";1;n$CODE a,c+12: GO TO 500
480 GO TO 450
500 LET a$="*""m"";1; "
510 PRINT AT 8,0;"Um das MC-Programm nun ";m$;" zu laden und zu starten,  ist f
olgendes einzugeben:"
520 PRINT '"      CLEAR ";a-1
530 PRINT    "      LOAD ";a$( TO q);""""";n$;""CODE"
540 PRINT "     RANDOMIZE USR ";a+c
550 PRINT "           (Blockverschiebung)"
560 PRINT "     RANDOMIZE USR X"
570 PRINT "      (X = Aufrufadresse des          eigentlichen Maschinen-
 programms)"
```

Der Header , und was man mit ihm alles machen kann !

Jedes normal abgespeichertes Programm besitzt einen Header , egal ob es ein BASIC-Prog. , Screen , Code oder Array ist . Jedes dieser Programme besteht aus einer Bytefolge welche in zwei Schritten geladen wird ; zuerst ein gleichmäßiger Signalton mit dem Header , und nach einer kleinen Pause wieder ein Signalton mit dem anschließenden Hauptteil . Da nach dem ersten Signalton schon der Programmname und -typ erscheint müßen diese im Header enthalten sein . Aber neben diesen Daten sind noch andere wichtige Werte in ihm verschlüsselt , die ich hier nun im einzelnem erklären werde .

Grundsätzlich besteht jeder Header aus 17 Bytes . Diese teilen sich folgendermaßen auf :

- Der Wert des 1.Bytes kennzeichnet den Typ des Programmes .
  00h entspricht einem BASIC-Prog. (LOAD"")
  01h " " Zahlenarray (LOAD""DATA x())
  02h " " Buchstabenarray (LOAD""DATA x$())
  03h " " SCREEN$ oder CODE (LOAD""CODE)
- Die nächsten 10 Bytes enthalten den Namen des Programmes .
- Das 12. und 13. Byte geben die Länge der gesammten abgespeicherten Bytefolge an .
- Das 14. und 15. Byte haben je nach Programmtyp verschiedene Aufgaben: Bei BASIC-Prog. geben sie die Startzeile an (wenn ohne LINE gesaved wurde hat das 15.Byte den Wert 80h) .
  Für die Arrays (Typ 01h und 02h) enthält das 15.Byte den Codebuchstaben des Arraynamens (z.B. ein mit DATA a() abgespeichertes Array hat den Codewert 61h im 15.Byte des Headers) .
  Programme mit dem Typencode 03h , also SCREEN$ oder CODE erhalten aus den beiden Bytes die Startaddresse der zu ladenden Bytefolge .
- Die zwei letzten , das 16. und 17. Byte werden nur bei BASIC-Prog. benutzt und geben die Länge des reinen BASIC-Programmes ohne den Variablen an .

Die folgende Tabelle stellt eine Übersicht des bisher erklärtem dar :

	BASIC-Prog.	Zahlenarray	Buchstabenarray	CODE/SCREEN
1. Byte	00h	01h	02h	03h
2.-11. Byte	Name	Name	Name	Name
12.+13. Byte	ges.Länge	ges.Länge	ges.Länge	ges.Länge
14.+15. Byte	Startzeile (o. 15. = 80h)	15. enthält Arrayname	15. enthält Arrayname	Startaddresse
16.+17. Byte	Länge des BASIC ohne Var.	ohne Bedeutung	ohne Bedeutung	ohne Bedeutung

Nach dieser trockenen Theorie wollen wir jetzt einmal mit dem Header experimentieren :

Zuerst eine etwas abgeänderte Fassung des "Knacker"-Prog. aus dem 1.Info mit dem wir einen Header laden können und dann aufgelistet dessen Inhalt auf dem Bildschirm erscheint :

```
10 CLEAR 32511: LET a=32511: DATA 17,17,0,175,55,221,33,16,127,205,86,
   5,48,242,201
20 FOR f=1 TO 15: READ g: POKE a+f,g: NEXT f
30 PRINT "Jetzt Tape starten !": RANDOMIZE USR (a+1)
40 FOR f=17 TO 33: PRINT f-16,PEEK (a+f): NEXT f
```

Jeder der die SOKA Nr.4 hat kennt das Prog. "Mr. COPY" und den Effekt beim laden des Headers : links oben steht "Bytes:" und in der Bildschirmmitte erscheint blinkend "Mr. COPY" . Dieses wurde durch Einfügen von Befehlswörtern (CHR$ 6-23) in die zehn Bytes ,des Headers , die den Namen des Programmes angeben , erreicht . Mit dem nächsten Prog. kann man diesen und andere Effekte mit dem Namen anstellen :

```
10 LET a$="": FOR f=1 TO 10: READ b: LET a$=a$+CHR$ b: NEXT f
20 DATA 22,11,5,239,198,203,226,73,84,33  (Effekt wie "Mr. COPY")
oder
20 DATA 8,8,8,8,8,8,8,16,7,170  (verschluckt bei CODE ganze Anzeige)
und dann SAVE a$ oder SAVE a$ CODE ...
```

Wer einmal den Wunsch hat nur den Teil eines BASIC-Prog. abzusaven ist mit dem untenstehendem Programm genau richtig bedinnt : Man gibt nur die erste und letzte abzusavende Zeilennummer ein , und schon können Sie ihren Programm-Ausschnitt auf Cassette aufnehmen .

```
10 CLEAR 32496: LET a=32496
20 INPUT "Startzeile : ",c: LET a2=INT (c/256): LET a1=c-a2x256
30 INPUT "Schlusszeile : ",b: LET b2=INT (b/256): LET b1=b-b2x256
40 INPUT "Name des Programmes",a$: IF LEN a$>10 THEN GOTO 40
50 FOR f=1 TO 10: IF LEN a$<f THEN POKE a+32+f,32: NEXT f: GOTO 70
60 POKE f+32+a,CODE a$(f): NEXT f
70 DATA 33,a1,a2,205,110,25,229,33,b1,b2,205,110,25,209,213,237,82,34,
   27,127,34,31,127,225,221,33,16,127,195,112,9,0
80 FOR f=1 TO 32: READ d: POKE a+f,d: NEXT f: POKE a+46,128
90 RANDOMIZE USR (a+1)
```

Bei der 48K-Version des Spectrum's können alle CLEAR-Werte und die Variable a um 30000 erhöht werden . Dann muß man in den DATA-Zeilen aber auch die Zahlenfolge...,16,127,...in...,64,244,...umgewandelt werden , so wie auch...,27,127,...in...,75,244,... und ...,31,127,... in ...,79,244,... .

Viel Spass und Erfog beim "Headern" wünscht euch Thomas Bertoldo !

Diagramme

Jeder weiß das durch Diagramme und Schaubilder jeder Bericht etwas ver = ständlicher wird . Da ja jeder von uns einen Spectrum , und viele auch einen Drucker dazu , besitzen , kann er mit der entsprechenden Software solche nützliche Schaublder erstellen . Die nötige Software wird in diesem Bericht vorgestellt ; was ihr noch beisteuern müßt sind die Daten aus denen dann die Diagramme erstellt werden . Und damit sind wir auch schon bei unserem erstem Programm :

Eingabe der vorhandenen Daten !

```
10 REM Diagramme  Teil 1  Eingaberoutine          von T.B.   08.84
20 BORDER 0: PAPER 0: INK 7: CLS
30 INPUT "Anzahl der Daten (max.20) ? ";x: IF x>20 THEN GO TO 30
40 DIM a(x): DIM a$(x): FOR f=1 TO x: PRINT f;".";TAB 5;
50 INPUT (f;". Datenname (max. 8 Buchst.) :")'b$: GO TO 59-LEN b$
60 INPUT ("Wert von ";b$)'a(f): PRINT b$,a(f): LET a$(f)=b$: NEXT f
```

Die Eingabe kann wie eins der folgenden Beispiele aussehen :

1. Beispiel : Seitenzahlen der Club-Infos

1.Info = 8	4.Info = 15	7.Info = 14	10.Info = 11
2.Info = 16	5.Info = 11	8.Info = 10	11.Info = 11
3.Info = 15	6.Info = 13	9.Info = 12	12.Info = 15

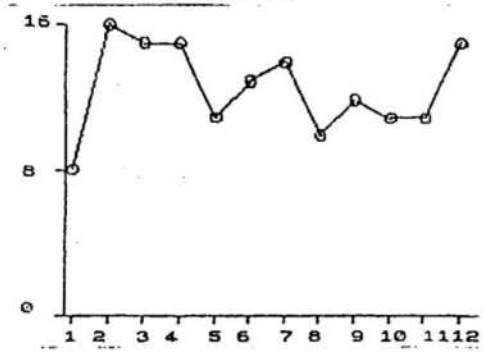
2. Beispiel : Inhalt der Club-Programmbibliothek (Stand 01.05.1984)

Action = 208 / Adv.,... = 67 / Brett,... = 31 / Anwend. = 85

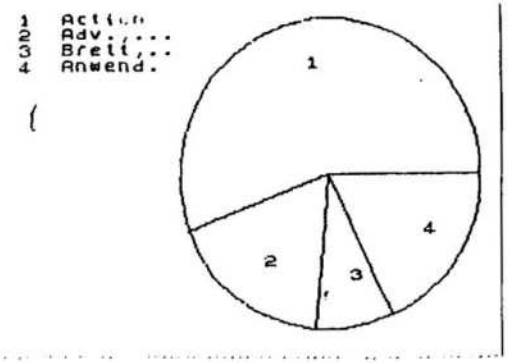
Nach der Eingabe enthält das Feld a(x) die gegebenen Werte und a$(x) die Bezeichnung des jeweiligen Wertes . Die nachfolgenden Programme arbeiten dann jeweils mit diesen Feldern , wie z.B. das folgende zur erzeugung einer farbigen Balkengraphik :

```
 99 REM Diagramme  Teil 2  Balkengraphik
100 CLS : LET b=0: FOR f=1 TO x: IF a(f)>b THEN LET b=a(f)
110 NEXT f: LET v=160/b: FOR f=1 TO x: PRINT a$(f)
120 FOR e=177-f*8 TO 183-f*8: INK ((f/2=INT (f/2))+4)
130 PLOT 90,e: DRAW v*a(f),0: NEXT e: INK 7: NEXT f: LET e=e-21
140 PLOT 90,e: DRAW 0,8: DRAW 160,0: DRAW 0,-8: PLOT 90+INT (b/2)*v,e
150 DRAW 0,8: PRINT ''TAB 11;0;TAB 20;INT (b/2);TAB 32-LEN STR$ b;b
```

Für ein Schaubild das den zeitlichen Verlauf eines bestimmten Wertes dar= stellen soll , eignet sich sehr gut ein Liniendiagramm . Das nebenstehende Diagramm zeigt ein solches Liniendiagramm welches die Seiten= zahl unseres heißgeliebten Infos über alle Ausgaben hinweg darstellt. Es wurde mit dem nächsten Programm er= zeugt .



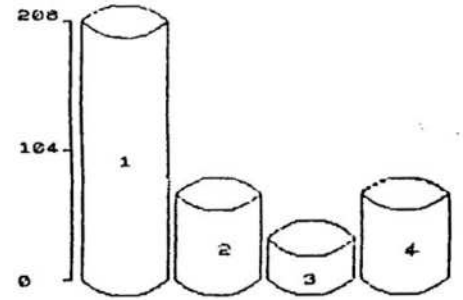

```
199 REM Diagramme  Teil 3
      Liniendiagramm
200 CLS : LET b=0: FOR f=1 TO x: LET b=b+(a(f)-b)*(b<a(f)): NEXT f
210 LET d=LEN STR$ b+1: LET v=155/b: LET c=(30-d)/(x-1): PLOT 255,15
220 DRAW -260+d*8,0: FOR f=0 TO x-1: PLOT 8*f*c+4+d*8,15: DRAW 0,-4
230 PRINT AT 21,f*c+d;f+1: NEXT f: FOR f=2 TO x: LET e=(f-2)*c*8+4+d*8
240 LET g=a(f-1)*v+15: CIRCLE e,g,3: PLOT e,g: DRAW c*8,a(f)*v-g+15
250 NEXT f: CIRCLE c*8+e,a(f)*v+15,3: PRINT AT 0,0;b;AT 0,0;b/2;AT 19,0;0
260 PLOT d*8,15: DRAW 0,155: DRAW -5,0: PLOT d*8-5,9: DRAW 5,0
```

Sehr wirksam zum veranschaulichen von prozentualen Aufteilungen irgendwelcher Summen ist das Kreisschaubild . Das hier gezeigte Bild ist eine Aufschlüsselung der gesammten Club-Programmbibliothek , unterteilt in vier Arten . Das untenstehende Programm machts möglich !

```
299 REM Diagramme  Teil 4  Kreisdiagramm
300 CLS : LET b=0: FOR f=1 TO x: LET b=b+a(f): NEXT f: LET b=b/100
310 DIM b(x): FOR f=1 TO x: LET b(f)=a(f)/b*.02*PI+b(f-1*(f>1)): NEXT f
320 FOR f=1 TO x: PLOT 172,88: DRAW COS b(f)*83,SIN b(f)*83
330 LET g=(172+COS (b(f)-(b(f)-b(f-1*(f>1))*(f>1))/2)*58)/8
340 LET h=21-(88+SIN (b(f)-(b(f)-b(f-1*(f>1))*(f>1))/2)*58)/8
350 PRINT AT f-1,0;f;TAB 3;a$(f);AT h,g;f: NEXT f: CIRCLE 172,88,83
```

Zum Abschluß noch eine Variante der Balkengraphik : Das Säulendiagramm , hier wieder mit dem Inhalt der Softwareliste von unserem Club . Erzeugt wurde sie mit dem letztem Programm der Diagramm-Serie :




```
399 REM  Diagramme  Teil 5  Säulendiagramm
400 CLS : LET b=0: FOR f=1 TO x: LET b=b+(a(f)-b)*(b<a(f)): NEXT f
410 LET c=LEN STR$ b+1: LET d=(30-c)/x*8: FOR f=1 TO x: LET e=140/b*a(f)
420 PLOT (f-1)*d+4+c*8,e+20: DRAW d-4,0,1.5: DRAW 4-d,0,1.5: DRAW 0,-e
430 DRAW d-4,0,1.5: DRAW 0,e: PRINT AT 20-e/16,c+d*(f-1)/8+d/16;f: NEXT f
440 PLOT c*8-6,20: DRAW 4,0: DRAW 0,140: DRAW -4,0: PLOT c*8-6,90
450 DRAW 4,0: PRINT AT 1,0;b;AT 10,0;b/2;AT 19,0;0
```

Ich hoffe die hier vorgestellten Programme helfen allen Spectrum-Besitzern um auch alles ins rechte Bild setzen zu können . Sollten trotzdem noch Fragen zu den hier vorgestellten Programmen auftauchen , kann man sich wie immer direkt an mich wenden .

Bis zum nächsten mal , euer

Thomas Bertoldo
Frankenbacherstr. 54
7100 Heilbronn